BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

REMARKS

Claims 1-53 are pending in the present application. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 102(b), Anticipation

The examiner has rejected claims 1-16, and 22-53 under 35 U.S.C. § 102(b) as being anticipated by Schutzman, United States patent number 5,822,780, ("*Schutzman*"). This rejection is respectfully traversed.

In rejecting the claims, the examiner stated:

> As to Claims 1,22,38, Schutzman teaches a system which including 'a data processing system for tracking relationships between programs and data' [see Abstract, col 5, line 32-47, fig 1b], Schutzman teaches data processing system, more specifically hierarchical storage management for database system where data files have close relationship with privileged programs such as detailed in fig 1b; 'receiving a file access request from a program' [col 7, line 43-50], Schutzman teaches request to access file related attributes is translated to unique identifier, then using this unique identifier, it checks to see if this particular file exists, if it does, it access the file as detailed in col 7, line 42-50, fig 5b; 'request is received at an operating system level' [col 9, line 2-11], operating system is integral part of Schutzman's teaching because Schutzman specifically suggests relationship between file management and kernel software for example Unix-based systems as detailed in col 5, line 45-47, further it is noted that Schutzman also teaches accessing file system operating as part of the computer system's operating system as detailed in col 9, line 2-6; 'storing an association between the file and the program' [col 6, line 12-24, col 9, line 12-15,fig 1c, fig 2], Schutzman specifically teaches relation between database file and standard database software as detailed in fig 1c, fig 2.

Office Action, dated March 1, 2004, pages 2-3.

Amended claim 1 reads as follows:

> 1. A method in a data processing system for tracking relationships between programs and data, the method comprising:
>
> receiving a file access request from a program, wherein the file access request is for a file and is received at an operating system level;

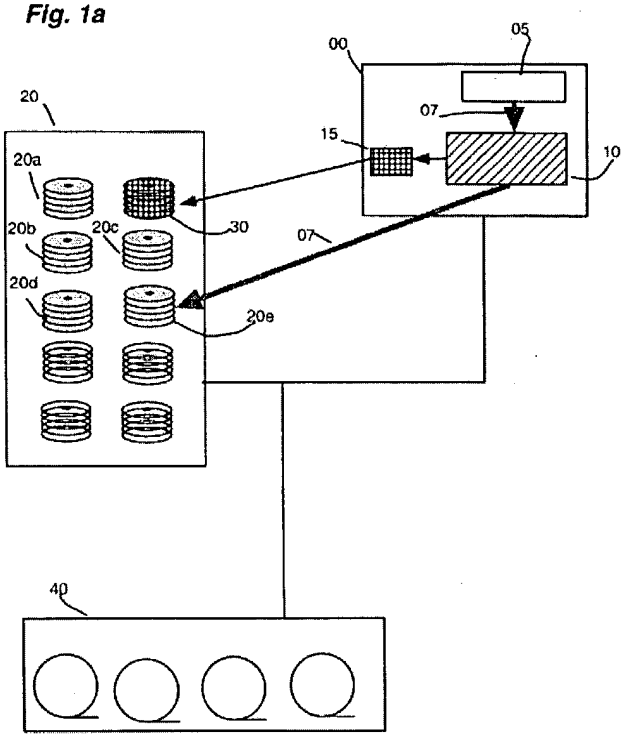
identifying an association between the file and the program requesting the file access in response to receiving the file access request; and

storing the association between the file and the program, wherein the association is used for subsequent accesses to the file such that a stored association is stored for each file for which file access is requested by the program.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102(b) only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Each and every feature of the presently claimed invention in amended claim 1 is not found in *Schutzman*.

In contrast, the cited reference is directed towards a hierarchical storage management system for database management systems that divides a database into regions (*Schutzman*, Abstract, lines 1-4). The system is show in Figure 1a below:

Fig. 1a



The system includes, in this figure, local disks 20 and tape robotics library 40. Element 40 also is referred to in *Schutzman* as tertiary storage 40. I/O calls to local disks 20 are intercepted but for a different purpose from the receiving step in claim 1. Claim 1 recites that the request for access causes an association to be identified, while *Schutzman* intercepts the call to identify where data is located, online in local disks or in tertiary storage.

Further, the sections of *Schutzman* cited by the examiner do not teach the features of the presently claimed invention in amended claim 1. Instead, these cited sections teach a method and apparatus for locating data in tertiary storage and moving this data from the tertiary storage to a location in the local disks when the requested data is located in the tertiary storage, rather than on the local disks.

For example, the examiner points to the following portion of *Schutzman*:

> A hierarchical storage management system for database management systems that divides a database logically into separately managed regions, with each region being described by an entry in a vector kept in a regions file. The region entry contains a time stamp of the last time the region was accessed, the staging identifier of the region if it has been migrated, the

base level backup staging identifier of the region if it has been baselined, and an indicator telling whether or not the region is resident online. Each managed region of the database is migrated to a separate staging file. When the database software issues a read or write input/output operation, the present invention sends the migration software a signal signifying this has occurred. The migration software of the present invention then updates the accessed time stamp, and checks to see if the region is resident. If it is not resident, it is staged in. The present invention creates and updates a migration candidate list ordered by date last accessed and region size. Both demand staging by the HSM, and user--initiated staging can then operate on the migration candidate list to migrate suitable files to tertiary storage.

Schutzman, Abstract. This section of the cited reference teaches a way to identify whether data is online, on local disks or whether the data is in a tertiary storage. If the data is not online, the data may be staged in, such as moved online to the local disks from the tertiary storage. No teaching is present for associating a relationship between programs and data.

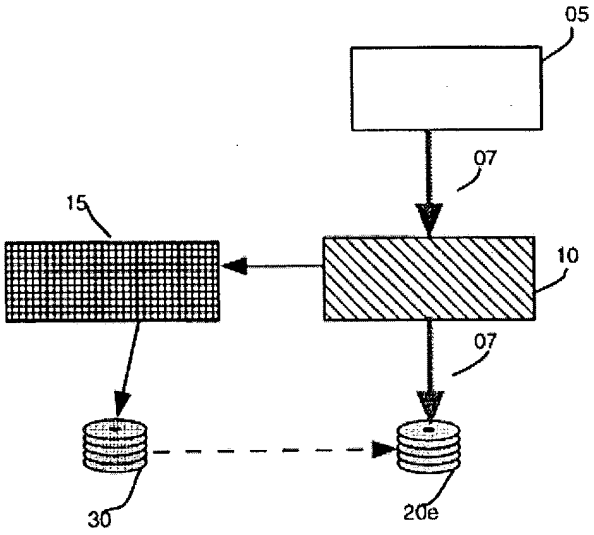
The next section cited by the examiner reads as follows:

In FIG. 1b, part of this is shown in a simpler diagram. Here, database software 05, which could be any type of software designed to access information on disks or databases, whether in a relational structure or not, is shown issuing an I/O request 07. Interceptor 10, is shown intercepting I/O request 07. In a preferred embodiment, interceptor 10 is a software program that is part of a file system that conforms to the proposed industry standard data management application programming interface specification (DMIG) prepared by the XOPEN standards organization. However, as will be apparent to those skilled in the art, any file system can be used as long as some way exists for a programmer to write a program to intercept I/O requests. In Unix-based systems for example, privileged programs can be written to intercept I/O requests made to the operating system's file management and kernel software.

Schutzman, column 5, lines 32-47. This portion of *Schutzman* teaches the use of database software to access information on disks or databases in which I/O requests may be intercepted. No teaching is present for identifying an association between the program requesting access to the file and the file as recited in claim 1.

The last section of *Schutzman* pointed to by the examiner is Figure 1b shown as follows:

Fig. 1b



This figure is described in the section cited by the examiner above with respect to intercepting I/O calls made to local disks. This figure does not teach or disclose identifying an association between a file and a program requesting the file. Instead, this figure is shown and described with respect to an interceptor, interceptor 10, that is used to intercept I/O requests. Migration software 15 receives an event generated by interceptor 10 and tracks activity on the database residing on local disks 20a-20e. Managed regions file 30 in Figure 1b is used to track the activity. Nowhere does this figure or the description of this figure teach or disclose an association with a file and a program requesting the file as recited in claim 1.

Thus, *Schutzman* does not teach tracking relationships between programs and data as believed by the examiner. Instead, the cited reference discloses a mechanism to locate data.

The examiner cites to the following portion of *Schutzman* for the step of storing an association between the file and the program:

> Now turning to FIG. 2, it can be seen that managed regions file 30 contains two vector entries 50e and 52e, pointing to managed regions 50 and 52 respectively, on disk 20e. In the schematic of FIG. 2, the contents

> of vector entry 50e are shown. As illustrated, each entry contains a last accessed timestamp 54, a staging id (sid) 56, a baseline id (blid) 58, and a residency indicator (res) 60. In the example shown for entry 50e, the last accessed timestamp is shown as Jan. 3, at 10:30 hours. The staging id 56 for entry 50e is 3. In a preferred embodiment, staging id 56 tells migration software 15 how to locate the data if it has been migrated. In this example, there is also a baseline id 58, with a value of 4.

Schutzman, column 6, lines 12-24. This cited section does not teach storing an association between the file and the program. Instead, this section of *Schutzman* teaches using an identifier to locate the data if the data is not located on the local disk, but on tertiary storage. This interpretation is evident when the cited section of *Schutzman* is read in context with the preceding section:

> In a preferred embodiment, once a request has been intercepted, interceptor 10 creates an event that signifies to migration software 15 that an interception has occurred. Migration software 15 operates as part of an HSM system in a preferred embodiment and creates and uses a managed regions file 30 to keep track of activity on the database residing on local disks 20a through 20e. When migration software 15 is notified of the occurrence of the interception, it reads the vector entry in the managed regions file 30 that covers that region of the database on disk 20e specified by the I/O request.
>
> Referring now to FIG. 1c, the HSM system of a preferred embodiment is shown. As seen here, migration software 15 includes an HSM aware application 15a, such as backup, a stage-in daemon 15b, a master daemon 15c, a migration service process 15e, and an HSM file system monitor 15f. A standard application, such as database software 05, makes an I/O request 07 which is intercepted by interceptor 10, operating as part of a filesystem 08 in the kernel 09 of the operating system controlling computer system 00. As seen in FIG. 1c, callbacks 15cb can cause work requests 15wr to be sent to migration service process 15e, as appropriate. HSM file system monitor 15f reads and writes data to local disks 20 through filesystem 08 as well, to satisfy migration requests.

Schutzman, column 5, line 54 – column 6, lines 11. The section teaches that requests for data located on local disks may be intercepted in *Schutzman* and the entry for the region covered by the request is read. The details of the information in these requests are described in the section cited by the examiner. These sections do not store associations between programs and files. Instead, these sections are used to locate data that has been migrated. No association between the program requesting the data and the data that is

stored in the section is cited by the examiner. The identity of the program does not matter and is not stored in association with the program in *Schutzman*, unlike amended claim 1.

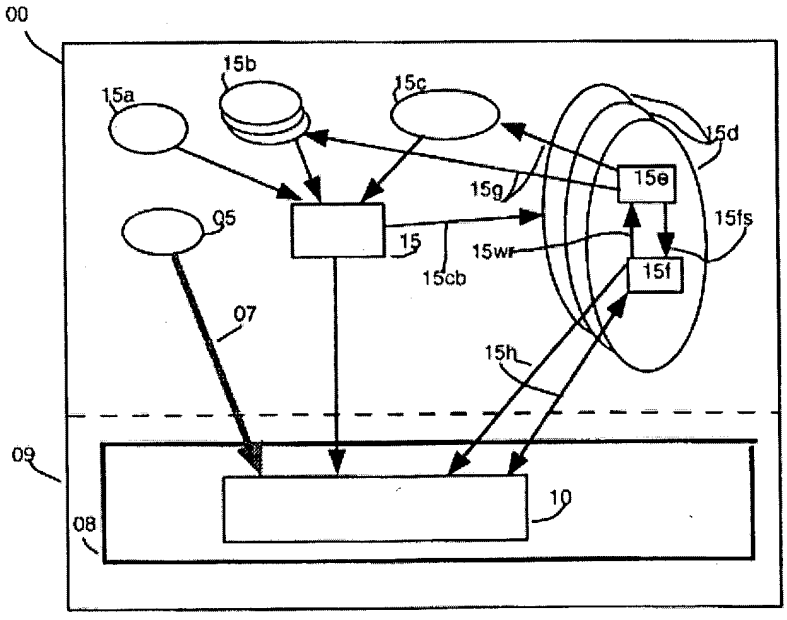
The examiner then cites to claim 1 of *Schutzman* as follows:

> a managed regions file residing on secondary storage media and having at least one entry in it corresponding to a physical area of the database, the entry including a last accessed timestamp and region attributes;

Schutzman, column 9, lines 12-15. This portion of *Schutzman* provides no teaching to store an association between a file and a program requesting the file as recited in claim 1. Instead, this portion of *Schutzman* teaches the use of a file that resides on a tertiary storage and an entry in the file that corresponds to a physical area in the database. No associations between files and programs are taught.

The examiner, also points to Figure 1c, which is shown as follows:

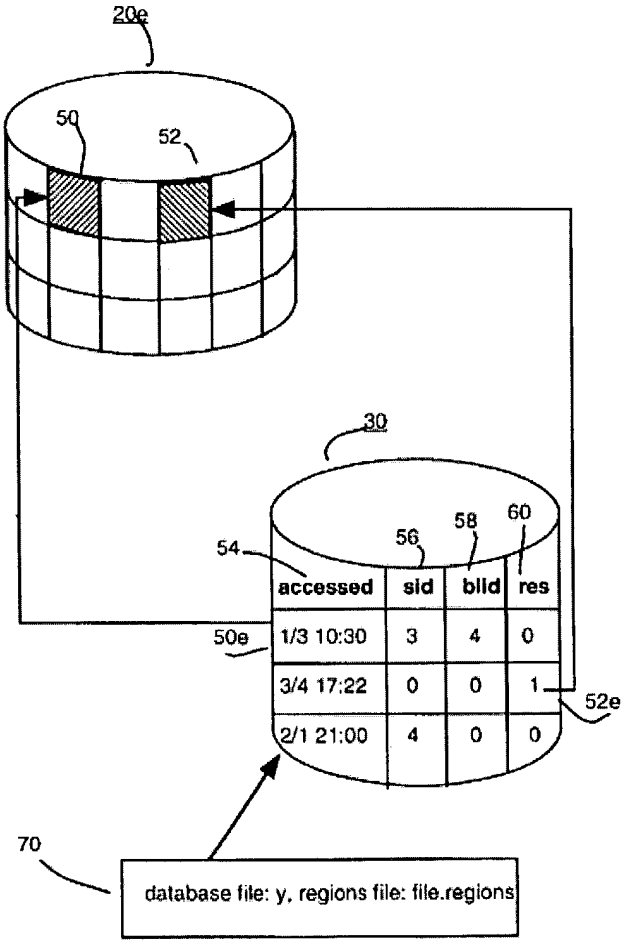
Fig. 1c



This figure provides no teaching for storing an association between a file and a program requesting the file. Instead, this figure is described in the specification of *Schutzman* for intercepting I/O requests.

The examiner also pointed to Figure 2. This figure is shown as follows:

Fig. 2



This figure shows a relationship between areas in a local disk, local disk 20e, and a managed region file, managed region file 30. Associations between a program requesting a file and the file are not stored in Figure 2. Instead, the cited figure from *Schutzman* shows including information about how to locate the file and other information about when the file was last accessed. However, an association between the file and the program requesting the file, as recited in claim 1, is not stored in managed region file 30.

Thus, this figure and the associated description does not teach or disclose saving associations between files and programs.

Independent claims 22 and 38 contain similar features. Thus these claims are not anticipated by *Schutzman*. The other claims are dependent claims and are patentable over *Schutzman* for the same reasons as the independent claims. Additionally, the dependent claims claim other additional combinations of features not suggested by the reference. Consequently, it is respectfully urged that the rejection of claims 1-16, and 22-53 under 35 U.S.C. § 102(b) have been overcome.

Furthermore, *Schutzman* does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. *Schutzman* actually teaches away from the presently claimed invention because it teaches a method and apparatus for managing data that is stored in a hierarchical fashion, such as on local disks and tertiary storage opposed to a method and apparatus for tracking relationships between programs and data as in the presently claimed invention. Absent the examiner pointing out some teaching or incentive to implement *Schutzman* in this manner, one of ordinary skill in the art would not be led to modify *Schutzman* to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify *Schutzman* in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the applicants' disclosure as a template to make the necessary changes to reach the claimed invention.

II. 35 U.S.C. § 103(a), Obviousness

The examiner has rejected claims 17-21 under 35 U.S.C. § 103(a) as being unpatentable over Schutzman, United States patent number 5,822,780, ("*Schutzman*") in view of Ogura, United States patent number 6,145,044, ("*Ogura*"). This rejection is respectfully traversed.

In rejecting these claims, the examiner has combined *Ogura* with *Schutzman*, using *Ogura* to provide the bus system, communication unit, memory, and processing unit in combination with the other features believed to be taught by *Schutzman*. As discussed above, *Schutzman* does not teach or suggest the features of the claimed

invention with respect to the other independent claims. As a result, a combination of *Schutzman* with *Ogura* would not reach the presently claimed invention.

Therefore, the rejection of claims 17-21 under 35 U.S.C. § 103(a) has been overcome.

III. Conclusion

It is respectfully urged that the subject application is patentable over the above-cited references and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: May 7, 2004

Respectfully submitted,

Duke W. Yee
Reg. No. 34,285
YEE & ASSOCIATES, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 367-2001
Attorney for Applicants